This listing of claims will replace all prior versions, and listings, of claims in the application:

<u>Listing of Claims</u>:

Claim 1 (currently amended): A system for ~~the~~ concurrent operation of plural computer applications, ~~each said computer application operating in its own virtual machine,~~ said system comprising:

    (a) <u>a computer storage medium including</u> a shared object space selectively connectable to each ~~said plural~~ <u>of a plurality of</u> computer ~~application~~ <u>applications</u>, said shared object space capable of storing<u>:</u>

        <u>(ii)</u> a plurality of ~~updateable~~ objects accessible to each <u>of</u> said plural computer ~~application when~~ <u>applications</u> connected to said shared object space; and

        (~~b~~ <u>ii</u>) a queue associated with said shared object space and capable of storing <u>a plurality of object</u> references ~~to individual said objects,~~ <u>each object reference</u> received from ~~at least~~ one of said plural computer applications and ~~capable of releasing said references stored in said queue to at least one of said plural computer applications~~ <u>identifying an individual object;</u> <u>and</u>

    <u>(b)</u> <u>at least one computer comprising at least two computer applications concurrently executing on respective virtual machines, a particular object updateable by one of said concurrently executing applications when said one application is connected to said shared object space and in control of an object reference from said queue identifying said particular object.</u>

Claim 2 (original): The system of claim 1 where said queue is a predefined type.

Claim 3 (original): The system of claim 1 where said queue is customized.

Claim 4 (original): The system of claim 1 where said queue is a "first-in-first-out" queue.

Claim 5 (original): The system of claim 1 where said queue is a "last-in-first-out" queue.

Claim 6 (currently amended):   The system of claim 1 where each said virtual machine ~~is a Java virtual machine~~ comprises a computer executable instruction in conformance with Java Virtual Machine Specification, said instruction executing on said computer.

Claim 7 (previously presented):   The system of claim 6 where said shared object space is connected to each said virtual machine through a Native Method Interface.

Claim 8 (original):   The system of claim 7 where said system includes a default directory with a native language library file.

Claim 9 (original):   The system of claim 1 where said shared object space is operably connectable to a non-object-oriented application.

Claim 10 (original):   The system of claim 9 where said non-object oriented program is a "C" program.

Claim 11 (previously presented):   The system of claim 1 where access to at least one of said plurality of objects by said plural computer applications is synchronized.

Claim 12 (currently amended):   The system of claim 1 where said ~~shared object space is operably connectable to a Sun Microsystems virtual machine~~ object reference further comprises a reference to one of said computer applications.

Claim 13 (original):   The system of claim 1 where said plural computer applications pertain to at least one of:
(a)   stock trading;
(b)   communications processing; and
(c)   internet services.

Claim 14 (previously presented):   The system of claim 1 where at least one of said plurality of objects is copy shared among said plural applications.

Claim 15 (previously presented): The system of claim 1 where at least one of said plurality of objects is direct shared among said plural applications.

Claim 16 (currently amended): A system for ~~the~~ concurrent operation of plural computer applications, ~~each said computer application operating in its own virtual machine,~~ said system comprising:

(a)   a computer storage medium including a shared object space selectively connectable to each ~~said plural~~ of a plurality of computer ~~application~~ applications, said shared object space capable of storing:

(i)   a plurality of ~~updateable~~ objects accessible to each of said plural computer ~~application when~~ applications connected to said shared object space; and

(~~b~~ ii)   a queue associated with said shared object space and capable of storing a plurality of object references ~~to individual said objects~~ , each object reference received from ~~at least~~ one of a first set of said ~~plural~~ computer applications and ~~capable of releasing said references stored in said queue to at least one of said plural computer applications~~ identifying an individual object; and

(b)   at least one computer comprising at least two computer applications concurrently executing on respective virtual machines, a particular object updateable by one of said concurrently executing applications belonging to a second set of said applications when said one application is connected to said shared object space and in control of an object reference from said queue identifying said particular object, said object reference identifying said particular object received in said queue from one of said computer applications of said first set ~~(c) said queue receiving said references from a first set of said applications and releasing said references to a second set of applications~~.

Claim 17 (original): The system of claim 16 where said queue is a predefined type.

Claim 18 (original): The system of claim 16 where said queue is customized.

Claim 19 (original):   The system of claim 16 where said queue is a "first-in-first-out" queue.

Claim 20 (original):   The system of claim 16 where said queue is a "last-in-first-out" queue.

Claim 21 (currently amended):   The system of claim 16 where each said virtual machine is a Java virtual machine comprises a computer executable instruction in conformance with a Java Virtual Machine Specification, said instruction executing on said computer.

Claim 22 (previously presented):   The system of claim 21 where said shared object space is connected to each said virtual machine through a Native Method Interface.

Claim 23 (original):   The system of claim 22 where said system includes a default directory with a native language library file.

Claim 24 (original):   The system of claim 16 where said shared object space is operably connectable to a non-object-oriented application.

Claim 25 (original):   The system of claim 24 where said non-object oriented program is a "C" program.

Claim 26 (previously presented):   The system of claim 16 where access to at least one of said plurality of objects by said plural computer applications is synchronized.

Claim 27 (currently amended):   The system of claim 16 wherein said shared object space is operably connectable to a Sun Microsystems virtual machine object reference further comprises a reference to said one computer application.

Claim 28 (original):   The system of claim 16 where said plural computer applications pertain to at least one of:

(a)   stock trading;

(b)   communications processing; and

(c)   internet services.

Claim 29 (previously presented):   The system of claim 16 where at least one of said plurality of objects is copy shared among said plural applications.

Claim 30 (original):   The system of claim 16 where at least one of said plurality of objects is direct shared among said plural applications.

Claim 31 (currently amended):   A system for the concurrent operation of plural computer applications, ~~each said computer application operating in its own virtual machine,~~ said system comprising:

(a)   a computer storage medium including a shared object space selectively connectable to each ~~said plural~~ of a plurality of computer ~~application~~ applications, said shared object space capable of storing:

(i)   a plurality of ~~updateable~~ objects accessible to each of said plural computer ~~application when~~ applications connected to said shared object space; and

(~~b~~ ii)   a queue associated with said shared object space and capable of storing a plurality of object references ~~to individual said objects~~, each object reference received from ~~at least~~ one of said plural computer applications and ~~capable of releasing said references stored in said queue to at least one of said plural computer applications~~; and

(~~e~~ b)   at least one computer comprising at least two computer applications concurrently executing on respective virtual machines, a particular object updateable by one of said concurrently executing applications when said one application is connected to said shared object space and in control of an object reference from said queue identifying said particular object, said object reference identifying said particular object received in said queue from said one application ~~said at least one application both storing said references in said queue and receiving said references from said queue.~~

Claim 32 (previously presented):   The system of claim 31 where said queue is a predefined type.

Claim 33 (previously presented):   The system of claim 31 where said queue is customized.

Claim 34 (previously presented):   The system of claim 31 where said queue is a "first-in-first-out" queue.

Claim 35 (previously presented):   The system of claim 31 where said queue is a "last-in-first-out" queue.

Claim 36 (currently amended):   The system of claim 31 where each said virtual machine is a Java virtual machine comprises a computer executable instruction in conformance with Java Virtual Machine Specification, said instruction executing on said computer.

Claim 37 (previously presented):   The system of claim 36 where said shared object space is connected to each said virtual machine through a Native Method Interface.

Claim 38 (previously presented):   The system of claim 37 where said system includes a default directory with a native language library file.

Claim 39 (previously presented):   The system of claim 31 where said shared object space is operably connectable to a non-object-oriented application.

Claim 40 (previously presented):   The system of claim 39 where said non-object oriented program is a "C" program.

Claim 41 (previously presented):   The system of claim 31 where access to at least one of said plurality of objects by said plural computer applications is synchronized.

Claim 42 (currently amended):   The system of claim 31 wherein said shared object space is

~~operably connectable to a Sun Microsystems virtual machine~~ <u>object reference further comprises a reference to said one computer application</u>.

Claim 43 (previously presented): The system of claim 31 where said plural computer applications pertain to at least one of:

(a) stock trading;

(b) communications processing; and

(c) internet services.

Claim 44 (previously presented): The system of claim 31 at least one of said plurality of objects is copy shared among said plural applications.

Claim 45 (previously presented): The system of claim 31 where at least one of said plurality of objects is direct shared among said plural applications.